

An adaptive mesh adjoint data assimilation method

F. Fang^{*}, M.D. Piggott, C.C. Pain, G.J. Gorman, A.J.H. Goddard

*Applied Modelling and Computation Group, Department of Earth Science and Engineering, Imperial College London,
Prince Consort Road, London SW7 2BP, UK*

Received 29 March 2005; received in revised form 5 January 2006; accepted 3 February 2006
Available online 3 March 2006

Abstract

An important issue in the numerical modelling of ocean dynamics is how to improve the predictive capabilities of models which include the representation of complex local flows around and over coastal topography. Four-dimensional (space and time) variational data assimilation represents a means to achieve this and can be realised using adjoint methods. An advantage of this approach is that it is able to fit a numerical simulation to observational data over both space and time. In this work, an adjoint method for an adaptive mesh forward model is developed and its correctness and feasibility verified. Here a forward and adjoint model, both based upon adaptive unstructured mesh finite element techniques, is tested by inverting boundary conditions in two-dimensional barotropic flow near a coastal headland. The inversion method uses independent adapting meshes for the forward and adjoint solution fields. This means that the transient characteristics of the different solution fields may be taken into account in the optimised computational meshes. The issues and advantages of this new data assimilation method are discussed. The computational efficiency of the method is presented. The robustness of the inversion (adjoint) model is tested by assimilating noisy and sparse observations into the model. It is seen that the inversion model can extract the mean observation information and produce good quality inversion results.

© 2006 Elsevier Ltd. All rights reserved.

1. Introduction

Complex geometry and nonlinear physical processes in ocean and coastal simulations require the use of high resolution numerical models capable of representing small scale features and large scale stratified flows around and over coastal topography. An adaptive unstructured mesh finite element model is being developed and applied to simulate such flows (Ford et al., 2004a,b). The model is fully three-dimensional, does not make the hydrostatic assumption, and employs mesh adaptivity to optimally focus computational resources when and where appropriate in the flow (Pain et al., 2001, 2005; Piggott et al., 2005). By using an unstructured mesh, the model can conform accurately to complex coastlines and topography, as well as allowing resolution to be adjusted to the requirements of the simulation as it evolves. For example, increased resolution may be

^{*} Corresponding author. Tel.: +44 0207 5949 322; fax: +44 0207 5949 341.
E-mail address: f.fang@imperial.ac.uk (F. Fang).
URL: <http://amcg.ese.imperial.ac.uk> (F. Fang).

employed in regions of high shear or flow separation from coastlines. As the model is able to respond to these emergent regions during the course of the calculation there is no need to impose a priori constraints on the mesh and therefore the solution, as is done with nested grid models for example.

Data assimilation efforts are primarily focused on reducing model errors introduced by uncertainties in the model, e.g. boundary conditions, initial conditions, wind friction and bottom drag coefficients, (Annan, 2001; Thacker and Long, 1988; Bennett and Thorburn, 1992; Giles, 1998; Martin et al., 2002). The assimilation of observations into a model is expected to significantly improve the predictive ability of the model, and reduce its sensitivity to these uncertainties. To perform the data assimilation an adjoint method is employed which can assimilate observations in space–time. The adjoint model is derived directly from the forward model equations. The adjoint calculation involves the minimisation of a cost function describing the misfit between the numerical solution and observations. This is typically performed using optimization techniques, such as the Nonlinear Conjugate Gradient (NCG), CON-strained MINimisation (CONMIN), Limited-Memory Quasi-Newton (LMQN), and Truncated Newton (TN) methods (Wang et al., 1998; Liu and Nocedal, 1989). The performance of these methods has been compared and analysed in for example (Zou et al., 1993; Morales and Nocedal, 2002; Liu and Nocedal, 1989).

Many people have studied 3/4D variational data assimilation e.g. Moore (1991), Tziperman et al. (1992), Wenzel et al. (2001), Daescu and Navon (2004), Weaver et al. (2003). An important step taken here is to investigate the introduction of mesh adaptivity into the adjoint model. Such an approach has the potential to provide more accurate results at a lower computational cost. However, it can introduce difficulties associated with preserving consistency which manifests itself in errors in the gradient calculation. Therefore the accuracy of the gradient calculation has to be checked. The application of this approach to two-dimensional flow past a Gaussian headland is presented here. The aim is to invert for open boundary conditions by assimilating observational data available within the computational domain. The discussion will focus on (a) the development of the mesh adaptive adjoint model; (b) the feasibility and advantages of mesh adaptivity in the forward and adjoint models, and subsequent computational efficiency; (c) the robustness of the inversion (adjoint) model, considered by assimilating sparse and noisy observations into the model; and (d) the importance of the nonlinearity in the momentum equations, which is addressed by comparing results from a simplified and the full adjoint models.

2. The forward model and mesh adaptivity

2.1. Underlying model equations

The ocean model utilised in this work makes use of the three-dimensional nonhydrostatic Boussinesq equations in a domain $\Omega \in \mathbb{R}^3$ with boundary Γ ,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + f \mathbf{k} \times \mathbf{u} = -\nabla p - \rho g \mathbf{k} + \nabla \cdot \boldsymbol{\tau}, \quad \nabla \cdot \mathbf{u} = 0, \quad (1)$$

where $\mathbf{u} \equiv (u, v, w)^T$ are the velocity components in the $\mathbf{x} \equiv (x, y, z)^T$ directions, p is the perturbation pressure, ρ is the perturbation density, f is the Coriolis parameter, g is the acceleration due to gravity, and $\mathbf{k} = (0, 0, 1)^T$. The stress tensor $\boldsymbol{\tau}$ is used to represent viscous terms and is defined in terms of the deformation rate tensor \mathbf{S} as

$$\tau_{ij} = 2\mu_{ij}S_{ij}, \quad S_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{1}{3} \sum_{k=1}^3 \frac{\partial u_k}{\partial x_k}, \quad (2)$$

with no summation over repeated indices. The horizontal viscosities (μ_{11}, μ_{22}) and vertical viscosity (μ_{33}) take constant values with the off-diagonal components of $\boldsymbol{\mu}$, defined by $\mu_{ij} = (\mu_{ii}\mu_{jj})^{1/2}$ see also (Tennekes and Lumley, 1972; Ford et al., 2004a).

For simplicity here density is assumed constant and thus the pressure p consists of hydrostatic $p_h(z)$ and nonhydrostatic $p_{nh}(x, y, z, t)$ components. The hydrostatic component of pressure balances exactly the constant buoyancy force and both terms are therefore dropped at this stage, (1) then takes the form

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + f \mathbf{k} \times \mathbf{u} + \nabla p_{\text{nh}} - \nabla \cdot \boldsymbol{\tau} = 0, \quad \nabla \cdot \mathbf{u} = 0. \quad (3)$$

The applications considered here make use of a barotropic model with a single layer in the vertical. The boundary and initial conditions for the forward model are given by

$$\mathbf{n} \cdot \mathbf{u}(x, t) = \mathbf{u}_r, \quad \mathbf{x} \in \Gamma \setminus \Gamma_{\text{out}}, \quad \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0, \quad (4)$$

where $\mathbf{u}_r \equiv \mathbf{u}_r(\mathbf{x}, t)$ and $\mathbf{u}_0 \equiv \mathbf{u}_0(\mathbf{x})$ are given functions, and \mathbf{n} represents the outward pointing unit normal.

In the work presented here the model is impulsively started via the imposed boundary conditions and $\mathbf{u}_0 \equiv 0$. The domain is that of a horizontal channel with an obstacle, in which case \mathbf{u}_r takes the value zero on lateral as well as upper and lower boundaries, i.e. no normal flow. Nonzero values ($\mathbf{u}_r = \mathbf{u}_b(\mathbf{x}, t)$) are imposed at an inflow boundary, signified by Γ_b . \mathbf{u}_b , is the quantity that will be inverted for in this work. The outflow boundary is signified by Γ_{out} and here no Dirichlet conditions are imposed. Finally, stress free traction boundary conditions of the form $\boldsymbol{\tau} \cdot \mathbf{n} - p\mathbf{n} = 0$ are also applied to all boundaries.

2.2. Numerical discretisation

A standard finite element discretisation of (1) (or (3)) represents solution variables as summations over a finite set of spatial basis functions. In this work, a tetrahedral mesh and piecewise linear basis functions are used for all variables, in certain notation this is referred to as a $P1 - P1$ element choice (Gresho and Sani, 1998). The discrete forms of the solution variables are then substituted into the governing equations and a Galerkin projection is taken by multiplying each equation by a basis function, integrating over the domain, and applying Green's theorem where appropriate to reduce the regularity required of basis functions. Discretisation of the temporal derivative, here via a Crank–Nicolson approach, yields a large system of coupled nonlinear equations to be solved. Nonlinearity is dealt with using Picard iteration and the momentum–pressure coupling is treated using a semi-implicit projection method (Gresho and Sani, 1998). The result is a nonsymmetric matrix system for momentum that is solved for using GMRES, and a symmetric positive definite matrix equation for pressure that is solved for using preconditioned conjugate gradients. Further details of the particular techniques employed here may be found in (Ford et al., 2004a). In addition, due to the well known spurious modes present with the $P1 - P1$ element choice (Gresho and Sani, 1998) a fourth-order pressure filter is employed to aid stability (Pain et al., 2001). Standard Petrov–Galerkin weightings (Hughes and Mallet, 1986) are also used to improve numerical stability in the presence of advection dominated flows.

2.3. Mesh adaptivity and error measures

2.3.1. Mesh optimisation

The mesh adaptivity and error measures technique used in this work adapts a mesh using a metric tensor which encodes local error estimates and is based on the curvature of the current solution fields, see (Power et al., 2006) for an alternative approach. The metric tensor is used to calculate the edge lengths of mesh elements. It is constructed (see below) such that an ideal edge length is unity when measured in metric space. This therefore gives a guide as to those elements which have edges that are too large or too small. Since the metric is dependent on both location and direction it is able to exhibit locally anisotropic information if the solution fields dictate this. Thus inhomogeneous and anisotropic meshes result from this approach. By defining an objective functional which is formed from the size and shape of the ‘worst’ element in the mesh, an optimisation technique is used to improve the overall quality of the mesh. That is, local operations on the mesh connectivity and node positioning are performed which aim to minimise this functional. The operations are performed on a three-dimensional tetrahedral mesh by the adaptivity method and include: edge collapsing/splitting; face to edge and edge to face swapping; edge to edge swapping; and local node movement or mesh smoothing. Constraints are imposed on these operations so as to preserve the integrity of nonplanar geometrical boundaries, e.g. the headland used later in this work. For further details see (Pain et al., 2001; Piggott et al., 2005; Gorman et al., 2006).

2.3.2. Definition of error metrics

The interpolation error resulting from a piecewise linear approximation of a smooth function ψ may be written in terms of the function's second derivatives or Hessian $\mathbf{H} \equiv \nabla^T \nabla \psi$. In particular, over a tetrahedral element the quantity

$$\varepsilon = \mathbf{v}^T |\mathbf{H}(\mathbf{x})| \mathbf{v} \quad (5)$$

gives a guide to this interpolation error at location \mathbf{x} and direction \mathbf{v} . Note that here the notation $|\cdot|$ corresponds to the absolute value of the matrix, defined by diagonalising the matrix and taking the absolute value of the eigenvalues, rather than the determinant. Taking the maximum of (5) over all locations and all vectors \mathbf{v} that lie within the element gives a bound on the magnitude of the interpolation error on this element. For more details see (Piggott et al., 2005) and the references therein.

Given a user defined error tolerance, $\hat{\varepsilon}$, a metric tensor may be defined by

$$\widehat{\mathbf{M}} = \frac{1}{\hat{\varepsilon}} |\mathbf{H}|. \quad (6)$$

To obtain this error tolerance everywhere, length scales in the mesh should therefore take the value unity when measured in the metric space defined by (6). The mesh optimisation technique described above is used to approximately achieve this. In practice, the Hessian must be approximated and here reconstructed from the piecewise linear solution fields. To simultaneously satisfy separate error bounds for separate solution fields a metric is obtained for each and these are then superimposed to yield a combined error metric. Finally, by suitably altering the definition of this metric computational constraints on the total number of nodes and maximum/minimum allowed element sizes and aspect ratios may be imposed. For further details see Pain et al. (2001).

In this work, the adjoint solution fields may vary largely in their magnitudes, typically taking a maximum value near detectors and containing some noise at lower values. This can lead to unwanted 'noise' in the resulting adapted mesh. Here successful use has been made of a metric that is weighted based on solution magnitudes. In effect this simply means that the error tolerance $\hat{\varepsilon}$ is increased where the solution fields take small values. This is achieved by defining the metric to be

$$\widehat{\mathbf{M}} = \frac{\min\{\text{tol}, |\psi|\}}{\hat{\varepsilon}} |\mathbf{H}|, \quad (7)$$

for solution field ψ , and a tolerance tol which allows one to impose an upper limit on the magnitude of the field after which this weighting procedure is no longer continued. For solution fields with small magnitude the imposed maximum element size effectively stops the metric going to zero, and hence avoids unreasonably large elements from being generated. Therefore the error metric tensor (6) is used to guide mesh adaptivity for the forward calculation, whereas (7) is used for the adjoint calculation.

3. Formulation of the adjoint equations

There are two standard techniques to derive an adjoint model. One is to obtain the discrete adjoint model directly from the discrete forward model, consistency with the discrete forward model is then guaranteed. Alternatively, the adjoint model may be derived in the continuum from the continuous forward model equations. Since the forward and adjoint model equations are similar, it is relatively straightforward to then construct a resulting adjoint numerical model from a forward model. The derivation of the adjoint equations can be performed using the Euler–Lagrange technique. Here, a Lagrangian functional \mathcal{L} is constructed from the weak form of the forward model (3) and an objective functional \mathcal{J} (11).

$$\mathcal{L}(\mathbf{u}, p_{\text{nh}}, \mathbf{m}, \mathbf{u}^*, p_{\text{nh}}^*) = \mathcal{J}(\mathbf{u}, \mathbf{m}) + \int_t \int_{\Omega} \left[\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + f \mathbf{k} \times \mathbf{u} + \nabla p_{\text{nh}} - \nabla \cdot \boldsymbol{\tau} \right) \cdot \mathbf{u}^{*\text{T}} + \nabla \cdot \mathbf{u} p_{\text{nh}}^* \right] d\Omega dt, \quad (8)$$

where \mathbf{u}^* and p_{nh}^* are adjoint variables, and \mathbf{m} are the control variables which may be model inputs, such as the boundary and initial conditions or uncertain parameters in the forward model, etc.

3.1. Tangent linear equations

By differentiating (3) with respect to control variables \mathbf{m} the following tangent linear equations are obtained:

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} + \mathbf{u} \cdot \nabla \bar{\mathbf{u}} + \bar{\mathbf{u}} \cdot \nabla \mathbf{u} + f \mathbf{k} \times \bar{\mathbf{u}} + \nabla \bar{p}_{\text{nh}} - \nabla \cdot \bar{\boldsymbol{\tau}} = 0, \quad \nabla \cdot \bar{\mathbf{u}} = 0, \quad (9)$$

where $\bar{\mathbf{u}}$ represents the variation of \mathbf{u} with respect to perturbations of the control variables. The tangent linear equations are used to determine the variation of the Lagrangian functional with respect to the control variables and leads to the formulation of the adjoint equations now presented.

3.2. Adjoint model

Differentiating the Lagrangian (8) with respect to the control variables, using (9), applying Green's theorem (in space and time) and making use of the relation

$$\Delta \mathcal{L} = \int_t \int_{\Omega} \nabla \mathcal{J} \Delta \mathbf{m} d\Omega dt,$$

allows one to derive the following adjoint equations:

$$-\frac{\partial \mathbf{u}^*}{\partial t} - \mathbf{u} \cdot \nabla \mathbf{u}^* - f \mathbf{k} \times \mathbf{u}^* - (\nabla \mathbf{u}^{*\text{T}}) \mathbf{u} - \nabla p_{\text{nh}}^* - (\nabla \cdot \boldsymbol{\tau})^* = (S_u, S_v, S_w)^{\text{T}}, \quad \nabla \cdot (-\mathbf{u}^*) = S_p, \quad (10)$$

with corresponding suitable boundary and initial conditions, as well as the gradient of the objective functional with respect to control variables. In (10), $[(\nabla \mathbf{u}^{*\text{T}}) \mathbf{u}]_i = \sum_{j=1}^3 u_j \partial u_j^* / \partial x_i$, and the adjoint viscous term takes the form

$$(\nabla \cdot \boldsymbol{\tau})_i^* = \sum_{j=1}^3 \frac{\partial}{\partial x_j} \left(\mu_{ji} \frac{\partial u_i^*}{\partial x_j} + \mu_{ij} \frac{\partial u_j^*}{\partial x_i} \right) - \frac{2}{3} \frac{\partial}{\partial x_i} \left(\sum_{l=1}^3 \sum_{k=1}^3 \mu_{kl} \frac{\partial u_l^*}{\partial x_k} \right),$$

with no summation over repeated indices. S_u, S_v, S_w, S_p are the source, or residual, terms at the detectors where the observations are available, i.e. the misfit between the numerical solution u, v, w, p and the corresponding observations, respectively,

$$S_u = \sum_{d=1}^D (u - u_{o,d}) \delta(x - x_{o,d}) \delta(y - y_{o,d}) \delta(z - z_{o,d}),$$

with the other terms being defined analogously. Here $\delta(\cdot)$ is the Dirac-delta function, $(x_{o,d}, y_{o,d}, z_{o,d})$, is the location of detector d , and $(u_{o,d}, v_{o,d}, w_{o,d})$ is the observed velocity at this detector.

The initial conditions for the adjoint model, corresponding to the problem considered here and described in Section 2.1, are given by

$$\mathbf{u}^*|_{t=T} = \mathbf{0},$$

where T is the end time of the simulation period. The boundary conditions are given by

$$\mathbf{n} \cdot \mathbf{u}^*(\mathbf{x}, t) = \mathbf{u}_r^*, \quad \mathbf{x} \in \Gamma \setminus \{\Gamma_b \cup \Gamma_{\text{out}}\},$$

where $\mathbf{u}_r^* \equiv \mathbf{u}_r^*(\mathbf{x}, t)$ is a given function, which here, as for the forward model, takes the value zero on lateral as well as upper and lower boundaries, i.e. no normal adjoint flow. Stress free boundary conditions are also applied to all boundaries.

The adjoint solution variables are used to calculate the gradient (see for example [Aleksiev and Navon, 2002](#); [Moore, 1991](#)) of the objective functional (11) for use in a nonlinear conjugate gradient minimisation procedure. The gradient of the objective functional at the boundary is given by

$$\left. \frac{\partial \mathcal{J}}{\partial u} \right|_{\Gamma} = -2uu^*.$$

For more details on the construction of the boundary conditions for the adjoint model and the gradient see (Gunzburger, 2003; Giles and Pierce, 2000).

4. Features of the forward and adjoint models

As discussed above, the adjoint model here is derived from the continuous forward equations, and then discretised. This distinguishes the approach from other discrete adjoint models (e.g. Marotzke et al., 1999; Vukicevic et al., 2001; Zou and Kuo, 1996; Zupanski et al., 2005; Daescu and Navon, 2004) formed directly by differentiating the discrete forward model, and often generated by automatic differentiation tools, e.g. TAMC (Giering and Kaminski, 1998) and ADIFOR (Bichof et al., 1992), which guarantee consistency between the discrete adjoint and discrete forward models. Advantages and disadvantages of both the continuous and discrete adjoint methods have been discussed in detail (Gunzburger, 2003; Sandu et al., 2003; Nadarajah and Jameson, 2000; Sirkes and Tziperman, 1997). As Sandu et al. (2003) states, there is no general rule to decide which approach should be used to implement the adjoint model. The motivation for the ‘differentiate then discretise’ approach used here is the ability to make good use of mesh adaptivity. This means that it will be possible to obtain high resolution simulations of local flow near complex coastlines and topography, as well as resolving regions near detectors for the adjoint calculation. In principle, by using the mesh adaptivity technique, the adjoint mesh will be optimised to best fit the adjoint solution/sensitivity, typically resulting in increased resolution in regions where the observational data is available. This approach needs to be checked for the consistency of the gradient calculation (see Section 7.2). The above issues are investigated here with a relatively simple barotropic model problem.

Gunzburger (2003) suggests that adjoint models are often best discretised using a different mesh to that used for the forward solution, so as to tailor the resolution, errors and stability properties of the calculation to best suit the adjoint solution features. The logical next step is to make use of a different ‘adapting’ mesh for this calculation. Sirkes and Tziperman (1997) discuss the two approaches in detail and note that some numerical schemes used for the forward modelling may not consider the features of the adjoint solution appropriately, which could therefore result in numerical artifacts in the adjoint calculation. However, Gunzburger (2003) also points out that because this approach leads to an inconsistent gradient, the downhill direction for the gradient may be an uphill direction for the cost function. In addition, the fact that freedom may be used in the error tolerance choice for the adjoint calculation opens up the possibility of accelerating the inversion procedure by making use of a hierarchy of increasingly fine meshes at early iterations. This should both reduce computational overhead, as well as aiding in the propagation of inversion information throughout the solution domain.

5. The objective functional and residuals

5.1. Definition and calculation of the objective functional

The objective of the optimisation, which is achieved by adjusting the uncertainties (control variables) in the model, is to minimise an objective functional. In this investigation, the objective functional is designed for inverting inflow velocity boundary conditions, and measures the misfit between the numerical solution and observed data, the continuous version of which takes the form

$$\mathcal{J}(\mathbf{u}, \mathbf{m}) = \frac{1}{2} \int_t \int_{\Omega} \sum_{d=1}^D (\mathbf{u} - \mathbf{u}_{o,d})^T w_d (\mathbf{u} - \mathbf{u}_{o,d}) d\Omega dt + \frac{1}{2} \int_t \int_{\Gamma_b} \lambda (\mathbf{m} - \hat{\mathbf{m}})^T (\mathbf{m} - \hat{\mathbf{m}}) d\Gamma dt. \quad (11)$$

Here d represents an index of detectors which yield observations, with D being the total number of such detectors $\mathbf{u}_{o,d}$ is the observed velocity vector at detector d ; \mathbf{m} are the control variables to be optimised using the adjoint method—here \mathbf{m} represents the unknown inflow velocity at the boundary \mathbf{u}_b ; w_d is a scalar weight

(a tensor could also be used to give different weights to different solution components) associated with the misfit between the numerical solution and observations at detector d , and can be used as a measure of ones confidence in the quality of the observations and may also vary with time; λ is a suitable scalar weighting for the penalty term in (11) which is used to avoid spurious functional minima (Gunzburger, 2003) and is defined in terms of the deviation between the updated boundary condition \mathbf{m} and the boundary condition from the previous optimisation iteration $\hat{\mathbf{m}}$; Ω is the computational domain and Γ_b is the section of its boundary where the unknown boundary conditions are located. A nonlinear conjugate gradient method is used here to minimise this functional (Bishop, 1995).

In the numerical model the possible mismatch between the location of detectors and solution collocation points (finite element nodes) must be dealt with. This mismatch will occur when the mesh varies in space and time, although it is feasible to lock mesh nodes to coincide with detectors or even to move in space–time with the detectors if applicable. To deal with this the detector data is ‘spread’ via convolution with a Gaussian function, $G \equiv G(\mathbf{x})$, to the nodes of the element that contains the detector, and perhaps further neighbouring elements. This also helps reduce numerical oscillations in the adjoint model (caused by point sources) near the detectors. These Gibbs oscillations occur despite the linearity of the adjoint model equations. Suppose

$$G \approx \sum_{j=1}^N G_j N_j(\mathbf{x}),$$

where $N_j \equiv N_j(\mathbf{x})$, $j = 1, \dots, N$, are the piecewise linear finite element basis functions, and G_j is the value of function G at node j , then

$$\int_{\Omega} G d\Omega \approx \int_{\Omega} \sum_{j=1}^N G_j N_j d\Omega = \sum_{j=1}^N G_j \int_{\Omega} N_j d\Omega = \sum_{j=1}^N G_j M_{ljj},$$

where $M_{ljj} = \sum_{i=1}^N M_{ij}$, $j = 1 \dots, N$, is the diagonal lumped mass matrix, and $M_{ij} = \int_{\Omega} N_i N_j d\Omega$ the mass matrix. Defining

$$W_{dj} = \frac{w_d G_j}{\sum_{j=1}^N G_j M_{ljj}}, \quad j = 1, \dots, N, \quad \text{and} \quad W_d = \sum_{j=1}^N W_{dj} N_j,$$

yields an approximation to G which satisfies $\int_{\Omega} W_d d\Omega = w_d$. The Gaussian function for detector d is defined by

$$G_d(\mathbf{x}) = \exp(-(\mathbf{x} - \mathbf{x}_d)^T (\mathbf{x} - \mathbf{x}_d) / 2L_d^2),$$

where x_d is the location of detector d , and L_d is the length scale over which the observation is spread.

The discrete form of the objective functional (11) can now be written as

$$\mathcal{J}(\mathbf{u}, \mathbf{m}) = \frac{1}{2} \sum_{k=1}^K \sum_{d=1}^D \sum_{j=1}^N M_{ljj}^k W_{dj}^k (\mathbf{u}_j^k - \mathbf{u}_{o,d}^k)^T (\mathbf{u}_j^k - \mathbf{u}_{o,d}^k) + \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^N \lambda \tilde{M}_{ljj}^k (\mathbf{m}_j^k - \hat{\mathbf{m}}_j^k)^T (\mathbf{m}_j^k - \hat{\mathbf{m}}_j^k), \tag{12}$$

where \mathbf{u} is the vector of numerical velocities approximating \mathbf{u} , \mathbf{u}_j^k represents the vector of 3 solution components at node j and time level k ; $\mathbf{u}_{o,d}^k \equiv (u_{o,d}^k, v_{o,d}^k, w_{o,d}^k)^T$ is the observed velocity at detector d ; superscripts represent the time level k with K being the total number of time levels; \mathbf{m}^k is the vector of discretised boundary conditions at time level k ; \tilde{M}_l^k is the lumped mass matrix on the mesh used at time level k restricted to the boundary, i.e.

$$\tilde{M}_{ljj}^k = \sum_{i=1}^N \tilde{M}_{ij}^k \quad \text{and} \quad \tilde{M}_{ij}^k = \int_{\Gamma_b} N_i N_j d\Gamma.$$

5.2. The residuals terms

The residuals terms S_u, S_v, S_w, S_p , in the adjoint model are the misfits between the numerical solution and observations. Again the location of source terms will not necessarily coincide with mesh nodes, and hence the same procedure of spreading used for the observation information is used here for the sources. The discrete form of the residual terms at node j and time level k therefore takes the form

$$S_{uj}^k = \sum_{d=1}^D W_{dj}^k (\mathbf{u}_j^k - \mathbf{u}_{o,d}^k),$$

with S_v, S_w, S_p defined analogously.

6. Implementation of the inversion procedure

The minimisation of the functional (12) with respect to the control variables \mathbf{m} is obtained here using a non-linear conjugate gradient method. This takes the form of a sequence of updates to the control variables of the form $\mathbf{m}_{i+1} = \mathbf{m}_i + \alpha_i \Delta \mathbf{m}_i$, where i is the iteration index. Given a conjugate gradient direction \mathbf{d}_i at iteration i the update may be written (Bishop, 1995),

$$\mathbf{m}_{i+1} = \mathbf{m}_i + \alpha_i \mathbf{d}_i, \quad (13)$$

where α_i is the step length in the search direction \mathbf{d}_i . The conjugate gradient direction is defined by $\mathbf{d}_{i+1} = -\mathbf{g}_i + \beta_i \mathbf{d}_i$, where the coefficient β_i can be written in the Polak–Ribiere form (Bishop, 1995)

$$\beta_i = \frac{\mathbf{g}_{i+1}^T (\mathbf{g}_{i+1} - \mathbf{g}_i)}{\mathbf{g}_i^T \mathbf{g}_i}. \quad (14)$$

The procedure is initialised with the conjugate gradient direction $\mathbf{d}_1 = -\mathbf{g}_1$, where \mathbf{g}_i is the gradient of the functional (12) calculated using the adjoint solution. The optimal step length α_i in the direction \mathbf{d}_i is obtained using a line search method which locates the functional minima along this direction. The line search method can be described as follows. First, given an initial three values $\alpha^{(1)}, \alpha^{(2)}, \alpha^{(3)}$, a quadratic polynomial is fit to functional values at the three points, the minimum of this parabola is then used as a new α which replaces one of the previous three. A new parabola is found and the procedure is repeated until the search step converges to the optimal α_i which minimises the functional.

As the meshes are adapted throughout a simulation, the total size of the control vector (here the number of mesh nodes on the inflow boundary) could vary. However, the dimensional sizes of $\mathbf{m}_i, \mathbf{d}_i$ and \mathbf{g}_i in (13) and (14) have to be the same for the optimisation procedure. Hence, here a Control Variable Mesh (CVM) is defined upon which the control variables are located during the line search procedure. The control variables \mathbf{m} from the current forward mesh are interpolated onto the CVM at every line search iteration. The CVM is then updated at every nonlinear conjugate gradient iteration. The previous gradient \mathbf{g}_i in (14) is therefore interpolated from the previous CVM to the current CVM. The adjoint solution is also interpolated from the adjoint mesh to the CVM when the conjugate direction in (13) is updated.

Note that here the forward and adjoint models will have the freedom to adapt to their own unique flow features, and therefore run on different adapted meshes. For the inversion of the control variables not only the flow variables but also the mesh must be stored, i.e. the node positions, at each time step. Due to constraints on memory check-pointing would also be used in practice. This information then needs to be interpolated between meshes as and when appropriate.

7. Example: flow past a headland

In this section attention is paid to (1) the correctness of the developed adjoint model; (2) the feasibility and advantages of the use of adaptive meshes for the forward and adjoint simulations; and (3) the computational efficiency and robustness of the adaptive adjoint model.

7.1. Problem formulation

The adaptive adjoint model is now applied to two-dimensional flow past a headland (a single layer of finite elements is used in the vertical). The computational domain is shown in Fig. 1. The domain spans 30,000 m in the east–west direction by 15,000 m in the north–south direction, with the southeast corner at the origin in this coordinate system. The headland is modelled by a southern boundary taking the form

$$y = d \exp(-(x - x_0)^2/2\sigma^2), \quad x_0 = 15,000 \text{ m}, \quad d = 3750 \text{ m}, \quad \sigma = 1000 \text{ m}.$$

In this test case the normal component of velocity at the inlet is inverted for using the adaptive mesh adjoint model. Pseudo-observational data is provided at several detector locations within the computational domain, and obtained by running the forward model forced by the expected boundary conditions. The expected velocity at the inlet ($x = 0$) is assumed to be $u = A \sin(2\pi t/T)$, where A is the maximum velocity (here 0.5 and 1.0 m have been used). Four detector positions are set to $x = 17,500 + 800(D - 1)$ and $y = 7500$ for $D = 1, 2, 3, 4$.

The assimilation time window is 3 h. The time step ranges between 20 and 100 s, whilst the amplitude of the boundary condition can vary between 0.5 and 1.0 m. The maximum and minimum adaptive mesh sizes are in the ranges 1000–5000 m and 100–500 m, respectively. The interpolation error imposed for the u and v velocity components ($\hat{\epsilon}$ in Section 2.3.2) is set to 0.05 for forward simulations and the measure (6) is employed; the interpolation error for the adjoint simulation is set to 0.005 and the measure (7) is employed with the cut-off tolerance (tol) set to 80.

7.2. Accuracy of the gradient calculation

In this test attention is paid as to whether the use of an inconsistent adjoint model (i.e. the ‘differentiate then discretise’ approach which can cause an inconsistency in the functional gradient) is feasible. Firstly the correctness of the developed adjoint model is verified, the comparison of results between a high resolution static and adaptive mesh models is performed in the following sections. The correctness of the adjoint model and code can be performed via a gradient check (Navon et al., 1992), i.e. the quantity

$$\Phi(\alpha) \equiv \frac{\mathcal{J}(\mathbf{m} + \alpha \mathbf{h}) - \mathcal{J}(\mathbf{m})}{\alpha \mathbf{h}^T \nabla \mathcal{J}(\mathbf{m})}, \tag{15}$$

is considered and it is verified that it is of order $1 + O(\alpha)$. In (15), \mathbf{h} is an arbitrary unit vector (such as $\mathbf{h} = \nabla \mathcal{J} / \|\nabla \mathcal{J}\|$), and \mathbf{m} are the control variables (here the normal velocity component at the inlet). For values of the scalar α that are small, it is expected that $\Phi(\alpha)$ is close to unity. To test this here the adjoint model has been applied to the above headland case (Fig. 1, problem 1B). The control variables \mathbf{m} are simply the discrete pointwise representation of $u_b = \sin(2\pi t/T)$, where $T = 3$ h. Fig. 2 shows the variation of the function $\Phi(\alpha)$

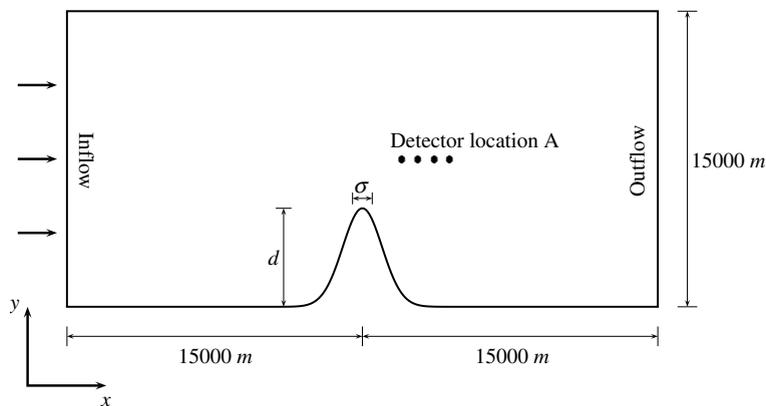


Fig. 1. The domain for two-dimensional flow past a Gaussian headland.

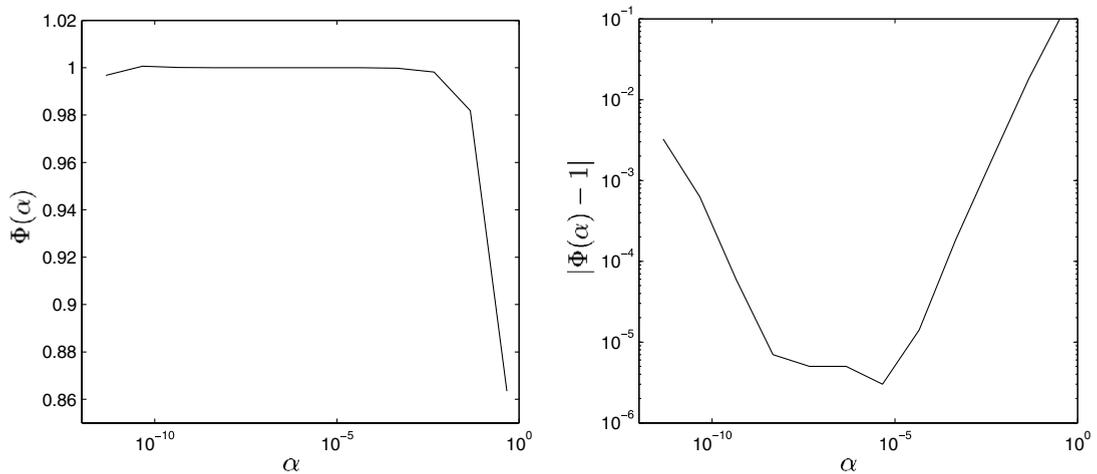


Fig. 2. The variation of $\Phi(\alpha)$ with respect to α .

with respect to α . It can be seen that as required the function $\Phi(\alpha)$ is close to unity when α varies between 10^{-4} and 10^{-7} . The variation of $\log_{10}|\Phi(\alpha) - 1|$ shows the characteristic ‘V’ shape, also as expected. It can thus be concluded, at least for this model problem set-up, that the adjoint model which is constructed here from the continuous forward model is correct and feasible.

7.3. Boundary condition inversion

The adaptive mesh adjoint model is now used to invert for the normal component of velocity at the inlet. Different computational set-ups are considered as described in Table 1. The inversion is started from the following initial guesses at the normal velocity component at the inlet:

$$u_b = -2.8 \left(\frac{t}{T} - \frac{1}{2} \right)^2 + 0.7 \text{ ms}^{-1}, \quad \text{for cases A,}$$

$$u_b = -4 \left(\frac{t}{T} - \frac{1}{2} \right)^2 + 1 \text{ ms}^{-1}, \quad \text{for cases B.}$$

The control vector can be written $\mathbf{m} = \mathbf{u}_b^T = (u_b^{1T}, \dots, u_b^{KT})$ where $u_b^{kT} = (u_{b,1}^k, \dots, u_{b,j}^k, \dots, u_{b,N_b^k}^k)$ and $u_{b,j}^k$ represents the normal velocity at inlet node j and time level k ; N_b^k is the total number of nodes at the inlet and at time level k ; K is the total number of time levels. The dimension of the control vector is $\sum_{k=1}^K N_b^k$, where N_b^k will change with time if the mesh adaptivity technique is employed. In theory, the control dimension must be the same as the number of observed data to be assimilated if the inversion problem is controllable and observable (Wunsch, 1996; Leredde et al., 1999). However, this is rarely achieved in realistic cases since observed data is often sparse.

The results are given in Fig. 3 where the comparison between the expected and optimal boundary conditions at the inlet ($x = 0$, $y = 7500$ m) are plotted. The spatial variation of the optimal normal velocity is also shown in Fig. 4. It can be seen that the maximum deviation of the normal velocity at the inlet is approximately 0.12 ms^{-1} at $t = 2000$ s, and the normal velocity is close to uniform during the rest of the period. The absolute velocity, plotted at the time level $t = 2500$ s, can be seen to decrease along the inlet (from 0 to 15,000 m), except at $y = 6000$ m which is affected by the headland.

The trend in the inverted boundary conditions is consistent with what is expected. The adapted mesh and flow field at an instance in time is shown in Fig. 5 for the forward model, and Fig. 6 for the adjoint model. As expected the meshes for the forward and adjoint models are completely different and are well suited to the forward and adjoint flow fields, respectively.

Table 1

Description of the various problem configurations used to compare fixed and adaptive mesh results

Problem	Amplitude	Time step (s)	Mesh
1A	0.5	100	Static meshes for both the forward and adjoint models, mesh size: 300 m
1B	1.0	100	
2A	0.5	100	Adapted mesh for the forward model, minimum and maximum adapted mesh sizes: 300 and 1500 m. Static mesh for the adjoint model, mesh size: 300 m
2B	1.0	100	
3A	0.5	100	Adapted mesh for the forward and adjoint models, minimum and maximum adapted mesh sizes: 300 and 1500 m
3B	1.0	100	
4A	0.5	20	Adapted mesh for the forward and adjoint models, minimum and maximum adapted mesh sizes: 100 and 1500 m
4B	1.0	20	
5B	1.0	100	Adapted mesh for the forward and adjoint models, minimum and maximum adapted mesh sizes: 150 and 1500 m

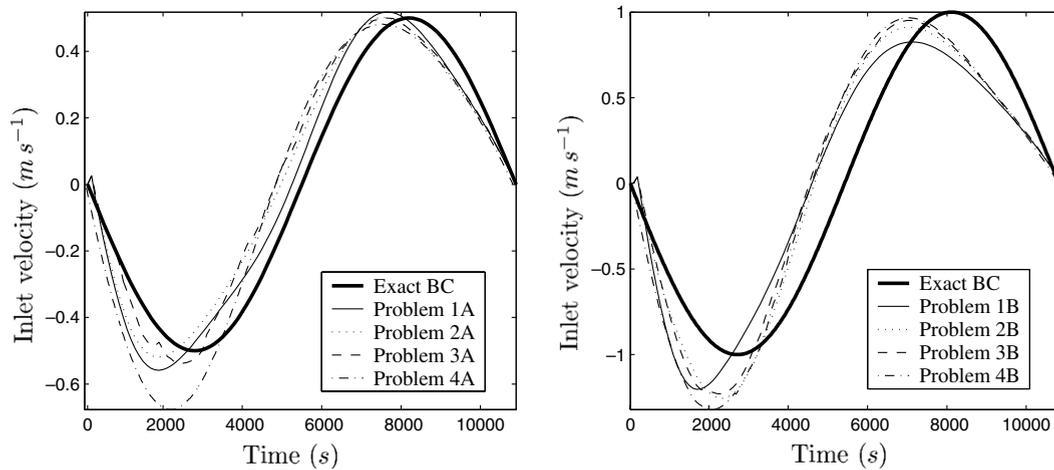


Fig. 3. Comparison of the inverted normal velocity component with and without mesh adaptivity. Left: for an exact solution with amplitude 0.5 m. Right: for an exact solution with amplitude 1.0 m.

7.4. Feasibility, advantages and computational efficiency of the adaptive adjoint model

There are two principal reasons that people doubt the feasibility of a continuum based adjoint model utilising mesh adaptivity. These revolve around errors introduced due to the interpolation between the forward and adjoint model meshes, as well as computational costs. The interpolation error may cause a serious problem, but it can be avoided by using a suitable adaptive technique and parameters (see Section 2.3). There is also concern that a lot of the detail computed with the adaptive mesh adjoint model is lost when computing the gradient. To help address this issue here a fixed fine mesh is imposed around the location where the gradient is calculated. For example, for the inversion of boundary conditions the mesh at the inflow boundary is locked in place so that the mesh resolution is not lost when the mesh is adapted. Note that although this prohibits mesh coarsening to a level beyond the initial mesh, mesh refinement is still permitted.

It is shown in Fig. 3 that the inverted results are quite close with and without mesh adaptivity. This means that the adaptive adjoint model can achieve the accuracy of a consistent adjoint model (i.e. the ‘discretise then

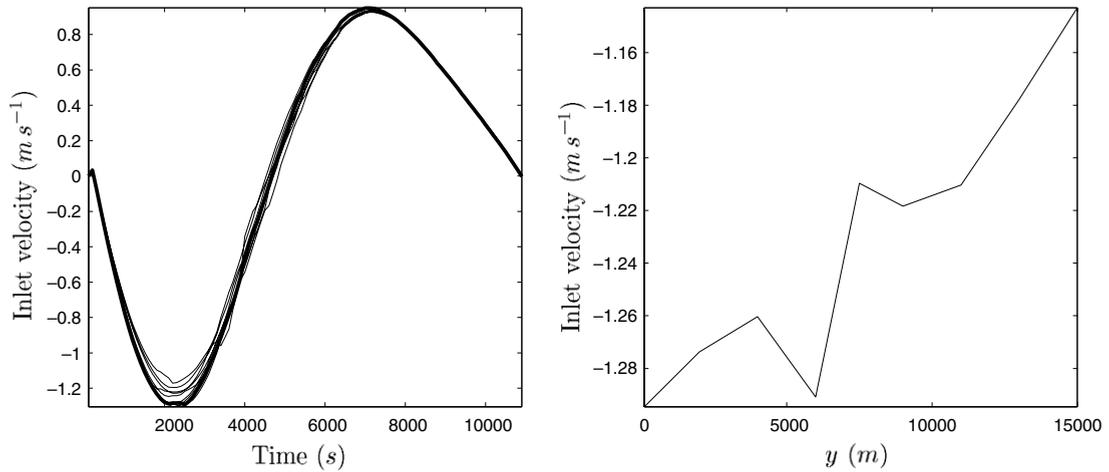


Fig. 4. Left: the temporal variation of the inverted boundary conditions at nine points along the boundary $x = 0$. Right: the inverted boundary condition at $t = 2500$ s as a function of the along boundary coordinate.

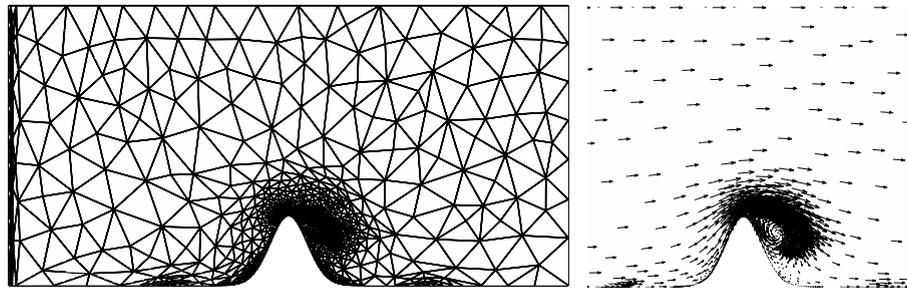


Fig. 5. The mesh and corresponding flow field around the headland for the forward calculation at time $t = 7500$ s for problem 3A.

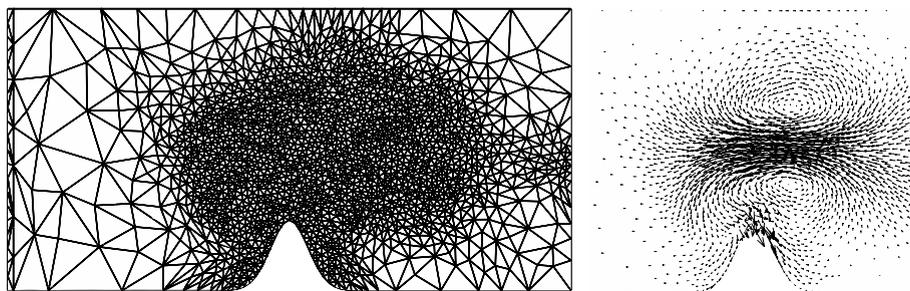


Fig. 6. The mesh and corresponding flow field around the headland for the adjoint calculation at time $t = 6400$ s for problem 3A.

differentiate' approach which guarantees consistency of the functional gradient), if it is assumed that the adjoint model with very fine static mesh (e.g. problems 1A and 1B) is close to the consistent adjoint model. Note here that the static mesh used in this comparison comprised 10,302 nodes; the adaptive forward mesh for problems 2 and 3 contains between 400 and 1600 nodes depending on the time level; the adaptive adjoint mesh for problem 3 contained in the region of 250–5000 nodes.

When seen in this light the results are very promising. For further comparison the number of nodes used during the simulation period with and without mesh adaptivity is plotted at the first and third NCG iterations (Fig. 7). It is demonstrated that as soon as the mesh is adapted the number of nodes significantly decreases to 10% of its original value in both the forward and adjoint calculations. During the whole simulation period the

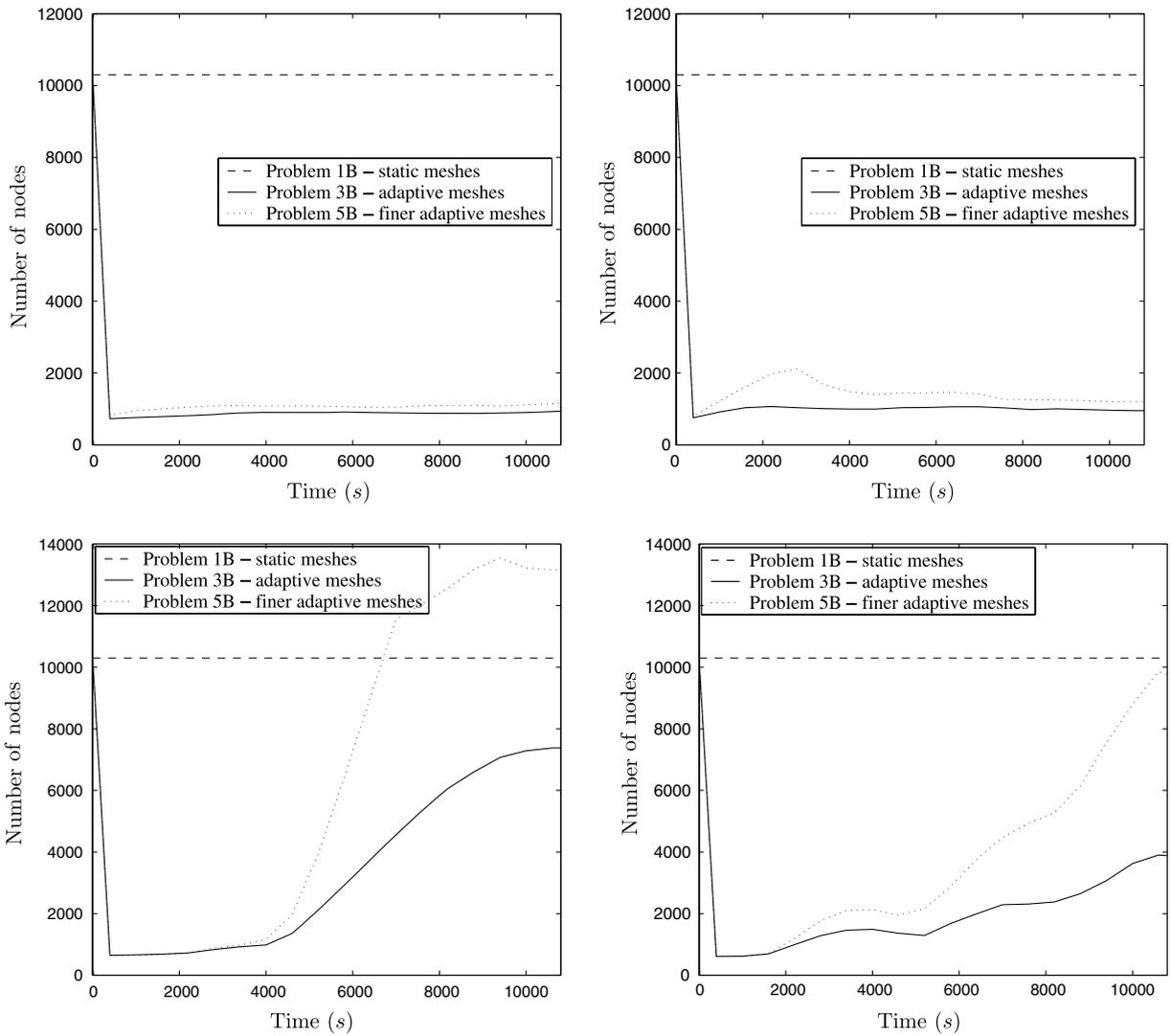


Fig. 7. Comparison of the number of nodes used during the simulation period with and without mesh adaptivity. Upper panels: the number of nodes in the forward calculation during the first (left) and third (right) iteration. Lower panels: the number of nodes in the adjoint calculation during the first (left) and third (right) iteration.

number of nodes for the case of adapting meshes is approximately 10% of that in the case of static meshes for the forward calculation, whilst mostly less than 50% for the adjoint calculation. A list of CPU times required is presented in Table 2. These simulations were performed on a dual Intel Xeon CPU 2.8 GHz workstation with 2 Gb of memory. It took 3–5 h to complete the inversion process when running the models using adaptive meshes, which is much less than the (16 h) required when using static meshes, whilst all the objective functionals decrease to 6.9–8.0% of their original values.

7.5. Robustness of the adjoint model

Observational data is often sparse and noisy, to examine the effect of this and test the robustness of the adjoint model white Gaussian noise is added into the pseudo-observational data at each detector in problem 3B. The amplitudes of the zero-mean noise vary randomly with time. The probability density of the noise is a Gaussian function. The amount of noise is increased by raising the values of the standard deviation (σ) of the probability density, here 0.05, 0.1, 0.3, 0.4, 0.5 and 1.0 are used. As an example, a set of noisy observational

Table 2

Comparison of the computational efficiency between the static (problem 1B) and adaptive mesh (problems 3B and 5B) models

Problem	CPU (h)	Normalised functional
1B—static mesh (mesh size: 300 m)	16.3 (In total) 6.2 (Iteration 1) 5.1 (Iteration 2) 5.0 (Iteration 3)	0.079
3B—adaptive mesh (mesh size: max 1500 m, min 300 m)	3.0 (In total) 1.1 (Iteration 1) 1.0 (Iteration 2) 0.9 (Iteration 3)	0.069
5B—adaptive mesh (mesh size: max 1500 m, min 150 m)	5.1 (In total) 1.8 (Iteration 1) 1.7 (Iteration 2) 1.6 (Iteration 3)	0.076

data (with standard deviation $\sigma = 0.5$) is shown in Fig. 8. The inverted velocity at the inlet ($x = 0, y = 7500$ m) is plotted in Fig. 9. It can be seen that even though noisy observational data is assimilated into the model, the inlet velocity can be recovered well. The adjoint model is able to smooth the noise and extract the mean information even with very noisy data.

7.6. The effect of extra (nonlinear) terms in the adjoint model

Comparing the forward model (3) with the left hand side of the adjoint model (10), it can be seen that they are virtually identical in structure except for the extra terms in the adjoint model which result from the nonlinear and viscous terms in the forward model, i.e. $(\nabla \mathbf{u}^{*T})\mathbf{u}$ and $(\nabla \cdot \tau)^*$. Assuming that the matrix of viscosities $\boldsymbol{\mu}$ is diagonal, then the adjoint viscous term reduces down to exactly the same form as the forward viscous term. This assumption is made here for simplicity. From the derivation of the adjoint model, the additional extra term, $(\nabla \mathbf{u}^{*T})\mathbf{u}$, resulted from the nonlinear term in the forward model. If the extra term can be ignored, then the forward model can be used and integrated backwards in time, with only very minor coding changes.

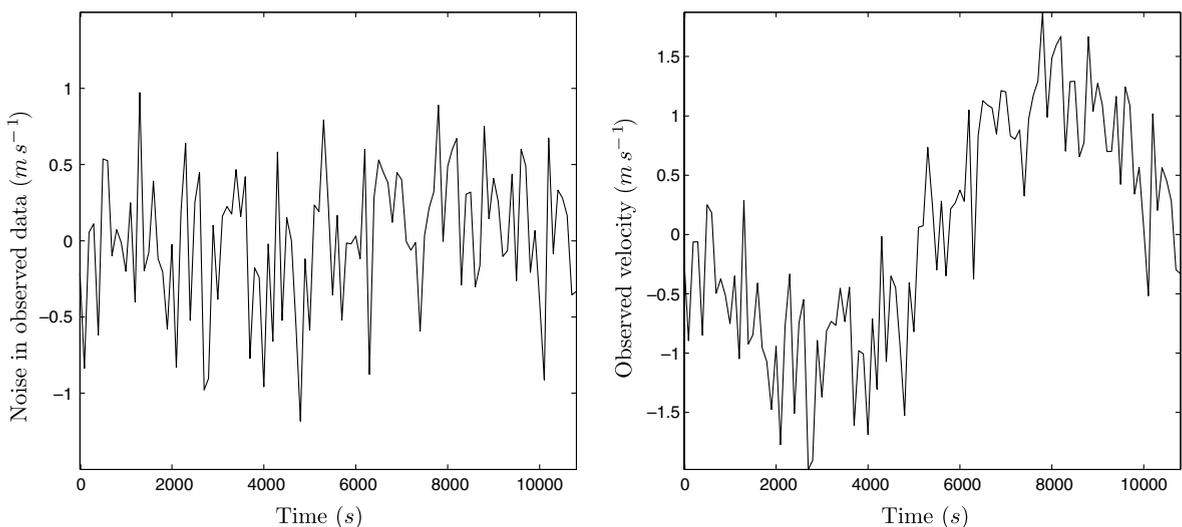


Fig. 8. Observational data with added white Gaussian noise for problem 3B. Left: white Gaussian noise data set with a standard deviation of $\sigma = 0.5$; Right: the pseudo-observational data at (17,500, 7500) m with the noise added.

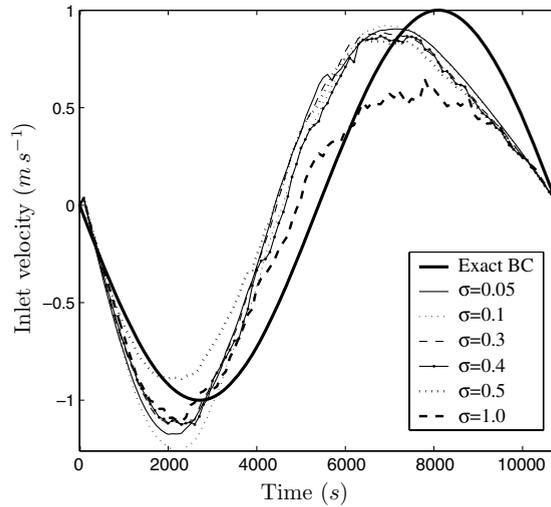


Fig. 9. Results from the examination of the effect of noisy observations in problem 3B for noisy data sets with different standard deviations. The inverted inlet velocity at point (0, 7500) m is plotted for different levels of noise in the observed data.

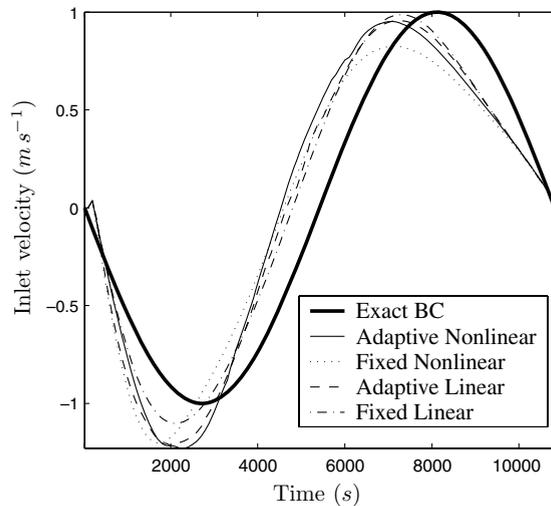


Fig. 10. Results from the examination of the effect of nonlinear terms. The inverted inlet velocity at point (0, 7500) m is plotted for the adjoint model with and without extra nonlinear contributions in the case of fixed (problem 1B) and adaptive (problem 3B) meshes.

In this section the effect of this extra term on the inversion solution is therefore investigated. The test case problems 1B and 3B have been performed both with and without the inclusion of the extra terms in the adjoint model. The subsequent solutions of the inversion are shown in Fig. 10. It can be seen that the difference between the inverted boundary conditions, with and without the extra terms in the adjoint model, is not as large as one might expect. This result suggests that to avoid writing adjoint codes and to make use of existing forward codes, simplified adjoint models may be used instead of the fully adjoint model in some situations, although in practice the inclusion of the additional terms is quite straightforward.

8. Conclusions and future work

An adaptive mesh ocean model has been developed to improve accuracy and efficiency in the simulation of complex flows. A corresponding adjoint model, also able to make use of adaptive mesh techniques, has also

been developed, as presented here. The adjoint model has been used to invert for pseudo-wave boundary conditions in a simple two-dimensional headland case. The accuracy of the gradient is verified and the robustness of the adjoint model is also investigated by assimilating sparse and noisy observations into the model.

The following conclusions can be drawn from the results obtained here. The use of an adaptive inversion model is indeed feasible and allows sufficient flexibility that the forward and adjoint meshes may be independently adapted according to the respective flow features of the forward and adjoint solution fields. The use of adaptive meshes can reduce both the computer memory requirement and CPU time (i.e. speed up the inversion process in the test case presented here). It is also demonstrated that the inversion model is robust/efficient enough to reproduce the boundary conditions even with sparse and noisy observations. The effect of the extra terms in the adjoint model, which come from the nonlinear terms in the forward model, on the inversion has been tested and found to be small for the problems presented here.

In future work, a sequence of hierarchically produced increasingly fine meshes are expected to accelerate and improve the quality of the overall inversion procedure. There are however still many open questions that remain to be investigated. For example, what are suitable error norms for use with inverse problems—this paper takes a first step towards answering this question. Goal based error measures which make use of both forward and adjoint solution information are being constructed (Power et al., 2006) and offer the possibility of optimising the accuracy of the inverse problem (defined by the data mismatch functional) and use the adjoint solution to help form the error norms that guide mesh adaptivity. A quantitative study of the improvements the adaptive mesh approach has over a static one is needed. The inversion of other types of model control variables is also being looked at. The use of a moving domain to model free surface problems is being considered, a first example of this is given in (Fang et al., 2005). The adjoint model will be further applied to three-dimensional realistic cases.

Acknowledgements

The authors would like to thank Professor I.M. Navon, for many helpful discussions. This work was carried out under funding from EPSRC (Grant GR/60898), and NERC (Grants NE/C52101X/1 and NE/C51829X/1). We would also like to acknowledge our colleagues on this collaborative research project at the University of Strathclyde: Dr. G.J.M Copeland and I.Y. Gejadze. We are very grateful to the anonymous reviewers for helping us to substantially improve this paper.

References

- Alekseev, A.K., Navon, I.M., 2002. On estimation of temperature uncertainty using the second order adjoint problem. *Int. J. Comput. Fluid Dyn.* 16 (2), 113–117.
- Annan, J.D., 2001. Hindcasting coastal sea levels in Morecambe Bay. *Estuarine, Coastal Shelf Sci.* 53, 459–466.
- Bennett, A.F., Thorburn, M.A., 1992. The generalized inverse of a nonlinear quasi-geostrophic ocean circulation model. *J. Phys. Oceanogr.* 22 (3), 213–230.
- Bichof, C., Carles, A., Corliss, G., Griewank, A., Hovland, P., 1992. ADIFOR-Generating derivative codes from Fortran programs. *Sci. Program* 1, 11–29.
- Bishop, C.M., 1995. *Neural Networks for Pattern Recognition*. Oxford University Press.
- Daescu, D.N., Navon, I.M., 2004. Adaptive observations in the context of 4D-Var data assimilation. *Meteorol. Atmos. Phys.* 85 (4), 205–226.
- Fang, F., Pain, C.C., Piggott, M.D., Gorman, G.J., Goddard, A.J.H., 2005. An adaptive mesh adjoint data assimilation method applied to free surface flows. *Int. J. Numer. Meth. Fluids* 47 (8–9), 995–1001.
- Ford, R., Pain, C.C., Piggott, M.D., Goddard, A.J.H., de Oliveira, C.R.E., Umbleby, A.P., 2004a. A nonhydrostatic finite-element model for three-dimensional stratified oceanic flows. Part I: Model formulation. *Mon. Weath. Rev.* 132 (12), 2816–2831.
- Ford, R., Pain, C.C., Piggott, M.D., Goddard, A.J.H., de Oliveira, C.R.E., Umbleby, A.P., 2004b. A nonhydrostatic finite-element model for three-dimensional stratified oceanic flows. Part II: Model validation. *Mon. Weath. Rev.* 132 (12), 2832–2844.
- Giering, R., Kaminski, T., 1998. Recipes for adjoint code construction. *Trans. Math. Software* 24, 437–474.
- Giles, M.B., 1998. On adjoint equations for error analysis and optimal grid adaptation. In: Caughey, D.A., Hafez, M.M. (Eds.), *Frontiers of Computational Fluid Dynamics 1998*. World Scientific, pp. 155–170.
- Giles, M.B., Pierce, N.A., 2000. An introduction to the adjoint approach to design. *Flow Turbul. Combust.* 65 (3–4), 393–415.
- Gorman, G.J., Piggott, M.D., Pain, C.C., de Oliveira, C.R.E., Umbleby, A.P., Goddard, A.J.H., 2006. Optimisation based bathymetry approximation through constrained unstructured mesh adaptivity. *Ocean Modell.* 12, 436–452.

- Gresho, P.M., Sani, R.L., 1998. Incompressible Flow and the Finite Element Method. John Wiley and Sons.
- Gunzburger, M.D., 2003. Perspectives in Flow Control and Optimization. SIAM.
- Hughes, T.J.R., Mallet, M., 1986. A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusion systems. *Comput. Methods Appl. Mech. Engrg.* 58, 329–336.
- Leredde, Y., Devenon, J.L., Dekeyser, I., 1999. Turbulent viscosity optimized by data assimilation. *Ann. Geophys.* 17, 1463–1477.
- Liu, D.C., Nocedal, J., 1989. On the limited memory BFGS method for large-scale optimization. *Math. Program.* 45 (3), 503–528.
- Marotzke, J., Giering, R., Zhang, K.Q., Stammer, D., Hill, C., Lee, T., 1999. Construction of the adjoint MIT ocean general circulation model and application to Atlantic heat transport variability. *J. Geophys. Res.* C12, 29529–29547.
- Martin, M.J., Bell, M.J., Nichols, N.K., 2002. Estimation of systematic error in an equatorial ocean model using data assimilation. *Int. J. Numer. Meth. Fluids* 40 (3-4), 435–444.
- Moore, A.M., 1991. Data assimilation in a quasi-geostrophic open-ocean model of the Gulf Stream region using the adjoint method. *J. Phys. Oceanogr.* 21 (3), 398–427.
- Morales, J.L., Nocedal, J., 2002. Enriched methods for large-scale unconstrained optimization. *Comput. Optim. Appl.* 21 (2), 143–154.
- Nadarajah, S.K., Jameson, A., 2000. A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimisation. AIAA-2000-0667.
- Navon, I.M., Zou, X., Derber, J., Sela, J., 1992. Variational data assimilation with an adiabatic version of the NMC spectral model. *Mon. Weath. Rev.* 120 (7), 1433–1446.
- Pain, C.C., Piggott, M.D., Goddard, A.J.H., Fang, F., Gorman, G.J., Marshall, D.P., Eaton, M.D., Power, P.W., de Oliveira, C.R.E., 2005. Three-dimensional unstructured mesh ocean modelling. *Ocean Modell.* 10 (1–2), 5–33.
- Pain, C.C., Umpleby, A.P., de Oliveira, C.R.E., Goddard, A.J.H., 2001. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Comput. Methods Appl. Mech. Engrg.* 190, 3771–3796.
- Piggott, M.D., Pain, C.C., Gorman, G.J., Power, P.W., Goddard, A.J.H., 2005. *h*, *r*, and *hr* adaptivity with applications in numerical ocean modelling. *Ocean Modell.* 10 (1-2), 95–113.
- Power, P.W., Pain, C.C., Piggott, M.D., Gorman, G.J., Fang, F., Marshall, D.P., Goddard, A.J.H., 2006. Adjoint goal-based error norms for adaptive mesh ocean modelling. *Ocean Modell.* This volume.
- Sandu, A., Daescu, D.N., Carmichael, G.R., 2003. Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: Part I - theory and software tools. *Atmos. Environ.* 37 (36), 5083–5096.
- Sirkes, Z., Tziperman, E., 1997. Finite difference of adjoint or adjoint of finite difference. *Mon. Weath. Rev.* 125, 3373–3378.
- Tennekes, H., Lumley, J.L., 1972. *A First Course in Turbulence*. MIT Press, Cambridge, MA.
- Thacker, W.C., Long, R.B., 1988. Fitting dynamics to data. *J. Geophys. Res.* 93 (C2), 1227–1240.
- Tziperman, E., Thacker, W.C., Long, R.B., Hwang, S.M., Rintoul, S.R., 1992. Oceanic data analysis using a general circulation model. Part II: A North Atlantic model. *J. Phys. Oceanogr.* 22 (12), 1458–1485.
- Vukicevic, T., Steyskal, M., Hecht, M., 2001. Properties of advection algorithms in the context of variational data assimilation. *Mon. Weath. Rev.* 129, 1221–1231.
- Wang, Z., Droegemeier, K., White, L., 1998. The adjoint Newton algorithm for large-scale unconstrained optimization in meteorology application. *Comput. Optim. Appl.* 10, 283–320.
- Weaver, A.T., Vialard, J., Anderson, D., Delecluse, P., 2003. Three- and four-dimensional variational assimilation with an ocean general circulation model of the tropical pacific ocean. Part ii: Physical validation. *Mon. Weath. Rev.* 131, 1379–1395.
- Wenzel, M., Schröter, J., Olbers, D., 2001. The annual cycle of the global ocean circulation as determined by 4D VAR data assimilation. *Prog. Oceanogr.* 48 (1), 73–119.
- Wunsch, C., 1996. *The Ocean Circulation Inverse Problem*. Cambridge University Press, Cambridge.
- Zou, X., Kuo, Y.H., 1996. Rainfall assimilation through an optimal control of initial and boundary conditions in a limited-area mesoscale model. *Mon. Weath. Rev.* 124 (12), 2859–2882.
- Zou, X., Navon, I., Berger, M., Phua, K., Schlick, T., Le Dimet, F., 1993. Numerical experience with limited-memory quasi-Newton and truncated Newton methods. *SIAM J. Optim.* 3 (3), 582–608.
- Zupanski, M., Zupanski, D., Vukicevic, T., Eis, K., Haar, T., 2005. Cira/csu four-dimensional variational data assimilation system. *Mon. Weath. Rev.* 133 (4), 829–843.